

Centos: Mise en oeuvre de Cobbler

Table of Contents

- [Centos: Mise en oeuvre de Cobbler](#)
- [Installation et configuration](#)
 - [Installation de cobbler](#)
 - [Mettre à jour le Système](#)
 - [Activer le référentiel EPEL](#)
 - [Installer cobbler](#)
 - [Configuration de cobbler](#)
 - [Mot de passe crypté par défaut](#)
 - [Server et Next_Server](#)
 - [Configuration du serveur DHCP](#)
 - [Démarrer et activer le service Cobbler](#)
 - [Vérifications et première synchronisation](#)
- [Importer une première distribution](#)
 - [Télécharger une image ISO](#)
 - [Exécuter l'importation](#)
 - [Vérification de l'import](#)
 - [Liste des objets](#)
 - [Détails d'objet](#)
 - [Créer un système](#)
- [Utilisation d'un fichier kickstart personnalisé](#)
 - [Création d'un fichier kickstart](#)
 - [Utilisation du fichier kickstart](#)
- [Utilisation d'un miroir yum local](#)
 - [Création d'un miroir Yum](#)
 - [Récupération des données](#)
 - [Utilisation des miroirs](#)

Cobbler est un serveur d'installation. Il permet d'installer facilement et automatiquement des systèmes d'exploitations, virtualisés ou non via pxe.

Cobbler est un projet fedora, il utilise donc de nombreuses technologies propres aux systèmes red-hat/Fedora (yum, rpm, kickstart...). Il est donc particulièrement adapté pour ces systèmes, mais des efforts sont en cours pour le rendre compatible avec debian notamment.

Installation et configuration

Installation de cobbler

Mettre à jour le Système

Il est vivement recommandé d'installer Cobbler sur un serveur et un système récemment mis à jour. Il est donc recommandé de mettre à jour les paquetages disponibles et de mettre à jour le système.

```
yum -y update
```

Activer le référentiel EPEL

```
yam -y install epel-release
```

Installer cobbler

Installer Cobbler avec les packages dépendants respectifs.

```
yum -y install cobbler cobbler-web dnsmasq syslinux pykickstart xinetd
```

Pour utiliser l'interface web, il faut également installer le paquet httpd.

Pour utiliser les outils de gestion de l'énergie, il faudra installer le paquet cman.



Pour éviter de nombreux problèmes avec SELinux, il est recommandé de le mettre en mode permissif et corriger les problèmes au fur et à mesure de leur apparition. Pour désactiver temporairement SELinux, exécuter la commande suivante `Setenforce 0`.

Ensuite, démarrer Cobbler et le serveur Web Apache et leur permettre de s'exécuter au démarrage.

```
systemctl start cobblerd  
systemctl enable cobblerd  
systemctl start httpd  
systemctl enable httpd
```

Configuration de cobbler

Avant de démarrer le service cobblerd, il faut modifier quelques éléments.

Les paramètres de cobbler/cobblerd sont stockés dans `/etc/cobbler/settings`. Ce fichier est

un fichier de données au format YAML, il faut donc faire attention lors de la modification de ce fichier car si celui-ci est mal formaté, cela empêchera cobbler/cobblerd de s'exécuter.

Mot de passe crypté par défaut

Ce paramètre contrôle le mot de passe root défini pour les nouveaux systèmes au cours du démarrage.

```
Default_password_crypted: "$1$bfI7WLZz$PxXetL97LkScqJFxnW7KS1"
```

Il faut modifier cela en exécutant la commande `openssl passwd -1` et en insérant le résultat dans la chaîne ci-dessus (veillez à enregistrer les guillemets):

- générer un mot de passe root chiffré et pour ce faire, exécuter la commande suivante.

```
# openssl passwd -1
Password:
Verifying - Password:
$1$j9/aR8Et$uovwBsGM.cLGcwR.Nf7Qq0
#
```

- mettre ce mot de passe root chiffré dans le fichier de paramètres `/etc/cobbler/settings` sous le paramètre `default_password_crypted`. Il faut également activer les fonctionnalités DHCP, DNS, PXE et TFTP de Cobbler en modifiant la valeur du paramètre de 0 à 1.

```
vi /etc/cobbler/settings

default_password_crypted: "$1$j9/aR8Et$uovwBsGM.cLGcwR.Nf7Qq0"
manage_dhcp: 1
manage_dns: 1
pxe_just_once: 1
next_server: YourServerIP
server: YourServerIP
```

Server et Next_Server

L'option `server` définit l'adresse IP qui sera utilisée pour l'adresse du serveur Cobbler. **NE PAS UTILISER 0.0.0.0**, car il ne s'agit pas de l'adresse d'écoute. Elle doit être définie sur l'adresse IP que les hôtes en construction utiliseront pour contacter le Serveur Cobbler activé pour des protocoles tels que HTTP et TFTP.

```
# default, localhost
server: 127.0.0.1
```

L'option `next_server` est utilisée pour DHCP/PXE en tant qu'adresse IP du serveur TFTP à partir duquel les fichiers d'amorçage du réseau sont téléchargés. En général, il s'agit de la même adresse IP

que celle de server.

```
# default, localhost
Next_server: 127.0.0.1
```

Configuration du serveur DHCP

Pour pouvoir démarrer PXE, on a besoin d'un serveur DHCP qui distribue les adresses et dirige le système d'amorçage sur le serveur TFTP, où il peut télécharger les fichiers d'amorçage réseau. Cobbler peut le gérer ce serveur, via le paramètre `manage_dhcp`:

```
# défaut, ne gère pas
manage_dhcp: 0
```

Gestion de DHCP par cobbler

Définir ce paramètre sur 1 pour que cobbler génère le fichier `dhcpd.conf` en fonction du fichier `dhcp.template` inclus dans cobbler.

- éditer `/etc/cobbler/dhcp.template` pour mettre à jour le sous-réseau du serveur DHCP.

```
$ vi /etc/cobbler/dhcp.template
```

Pour la plupart des utilisations, il vous suffira de modifier ce bloc:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
    option routers                192.168.1.1;
    option domain-name-servers   192.168.1.210,192.168.1.211;
    option subnet-mask           255.255.255.0;
    filename                     "/pxelinux.0";
    default-lease-time           2.8.0;
    max-lease-time               43200;
    next-server                   $next_server;
}
```



Ne pas modifier la ligne `next-server $next_server;`, car c'est ainsi que le paramètre `next_server` est intégré à la configuration. Ne rien modifier également après cette ligne:

```
#for dhcp_tag in $dhcp_tags.keys():
```

- mettre à jour la plage d'adresses IP dans le fichier `/etc/cobbler/dnsmasq.template`:

```
vi /etc/cobbler/dnsmasq.template  
  
dhcp-range=192.168.10.80,192.168.10.140
```

Gestion de DHCP en dehors de cobbler

Pour configurer le serveur dhcp, il faut rajouter ces deux lignes dans le fichier /etc/dhcp3/dhcpd.conf:

```
next-server ip-du-serveur-cobbler  
filename "pxelinux.0"
```

Ces lignes indiquent l'IP du serveur qui assure le service PXE et le nom du fichier à télécharger sur ce serveur.

Démarrer et activer le service Cobbler

Une fois les paramètres mis à jour, on peut démarrer le service.

Lorsque le système utilise systemctl pour gérer les services:

```
$ systemctl start cobblerd.service  
$ systemctl enable cobblerd.service  
$ systemctl status cobblerd.service
```

Ou le script init standard

```
$ service cobblerd start  
$ chkconfig cobblerd on  
$ service cobblerd status
```

Si tout s'est bien passé, on doit voir la sortie de la commande status comme ceci:

```
cobblerd.service - Cobbler Helper Daemon  
    Loaded: loaded (/lib/systemd/system/cobblerd.service; enabled)  
    Active: active (running) since Sun, 17 Jun 2012 13:01:28 -0500;  
1min 44s ago  
    Main PID: 1234 (cobblerd)  
    CGroup: name=systemd:/system/cobblerd.service  
            └─ 1234 /usr/bin/python /usr/bin/cobblerd -F
```

Vérifications et première synchronisation

Maintenant que le service cobblerd est opérationnel, il faut vérifier si des problèmes se sont produits. La commande `check` de Cobbler fera quelques suggestions, mais il est important de se rappeler que ce ne sont principalement que des suggestions et qu'elles ne sont probablement pas essentielles pour les fonctionnalités de base. Sous iptables ou SELinux, il est important de passer en revue tous les messages concernant ceux que le contrôle peut signaler.

```
$ cobbler check
The following are potential configuration items that you may want to fix:

....
....

Restart cobblerd and then run 'cobbler sync' to apply changes.
```

Lorsqu'on applique l'une des suggestions, telles que l'installation de packages supplémentaires, la modification de la configuration, etc., il faut redémarrer le service cobblerd.

Une fois que l'examen le résultat de “cobbler check” est terminé, il faut faire une première synchronisation. Ce n'est pas critique, mais un échec de la synchronisation correcte à ce stade peut révéler un problème de configuration.

```
$ cobbler sync
task started: 2012-06-24_224243_sync
task started (id=Sync, time=Sun Jun 24 22:42:43 2012)
running pre-sync triggers
...
rendering DHCP files
generating /etc/dhcp/dhcpd.conf
rendering TFTP files
generating /etc/xinetd.d/tftp
cleaning link caches
running: find /var/lib/tftpboot/images/.link_cache -maxdepth 1 -type f -
links 1 -exec rm -f '{}' ';'
received on stdout:
received on stderr:
running post-sync triggers
running python triggers from /var/lib/cobbler/triggers/sync/post/
running python trigger cobbler.modules.sync_post_restart_services
running: dhcpd -t -q
received on stdout:
received on stderr:
running: service dhcpd restart
received on stdout:
received on stderr:
running shell triggers from /var/lib/cobbler/triggers/sync/post/
running python triggers from /var/lib/cobbler/triggers/change/
```

```
running python trigger cobbler.modules.scm_track
running shell triggers from /var/lib/cobbler/triggers/change/
TASK COMPLETE
```

En supposant que tout se soit bien passé et qu'aucune erreur n'ait été signalée, on peut passer à l'étape suivante. .

Importer une première distribution

Cobbler automatise l'ajout de distributions et de profils via la commande `cobbler import`, qui permet (généralement) de détecter automatiquement le type et la version de la distribution importée et de créer (un ou plusieurs) profils avec les paramètres appropriés.

Télécharger une image ISO

Pour importer une distribution, on a besoin d'un fichier ISO de la distribution



Il faut utiliser une image DVD complet et non un fichier ISO Live CD.

Une fois le fichier téléchargé, il faut le monter quelque part:

Créer un répertoire, copier le fichier ISO de CentOS 7 et exécuter les commandes ci-dessous pour monter l'image ISO.

```
mkdir /mnt/iso
mount -o loop,ro /path/to/isos/Fedora-Server-dvd-x86_64-28-1.1.iso /mnt/iso/
```

Exécuter l'importation

On est maintenant prêt à importer la distribution. Les arguments nom et chemin sont les seules options requises pour l'importation:

```
cobbler import --arch=x86_64 --path=/mnt/iso --name=fedora28
```

L'option `--arch` n'a pas besoin d'être spécifiée, car elle sera normalement détectée automatiquement.

On peut aussi utiliser une url :

```
cobbler import --path=rsync://servergoeshere/path/to/distro --name=distro
```



si des erreurs de signature surviennent lors de l'importation, exécuter la commande ci-dessous pour résoudre le problème.

```
cobbler signature update
```



Cobbler utilise beaucoup le répertoire `/var`, qui regroupe tous les fichiers de la distribution et du référentiel, on aura besoin de 5 à 10 Go d'espace libre par distribution à importer.

Vérification de l'import

La commande `import` crée généralement au moins une paire `distro/profil`, qui porte le même nom que celui indiqué ci-dessus. Dans certains cas (par exemple, lorsqu'un noyau basé sur `xen` est trouvé), plusieurs paires `distro/profil` seront créées.

Liste des objets

Si aucune erreur n'a été signalée lors de l'importation, on peut afficher des détails sur les distributions et les profils créés lors de l'importation.

```
$ cobbler distro list
```

```
$ cobbler profile list
```

Détails d'objet

La commande `report` affiche les détails des objets dans Cobbler:

```
$ cobbler distro report --name=fedora28-x86_64
Name                               : fedora28-x86_64
Architecture                       : x86_64
TFTP Boot Files                    : {}
Breed                              : redhat
Comment                            :
Fetchable Files                     : {}
```

```
Initrd                : /var/www/cobbler/ks_mirror/fedora28-  
x86_64/images/pxeboot/initrd.img  
Kernel                : /var/www/cobbler/ks_mirror/fedora28-  
x86_64/images/pxeboot/vmlinuz  
Kernel Options        : {}  
Kernel Options (Post Install) : {}  
Kickstart Metadata    : <<inherit>>  
Management Classes    : []  
OS Version            : fedora28  
Owners                : ['admin']  
Red Hat Management Key : <<inherit>>  
Red Hat Management Server : <<inherit>>  
Template Files         : {}
```

Comme on peut le voir ci-dessus, la commande `import` remplit automatiquement plusieurs champs, tels que `breed`, `OS version` et l'emplacement des fichiers `initrd/kernel`.



Certains champs sont définis sur `<< inherit >>`. Cela signifie qu'ils utiliseront le paramètre par défaut (trouvé dans le fichier de paramètres) ou (dans le cas de profils, de sous-profils et de systèmes). Utilise tout ce qui est défini dans l'objet parent.

Créer un système

Maintenant qu'on a une distribution et un profil, on peut créer un système. Les profils peuvent être utilisés pour démarrer PXE, mais la plupart des fonctionnalités de `cobbler` tournent autour des objets système. Plus on donne d'informations sur un système, plus il le fera automatiquement.

Tout d'abord, créer un objet système en fonction du profil créé lors de l'importation. Lors de la création d'un système, le nom et le profil sont les deux seuls champs obligatoires:

```
$ cobbler system add --name=test --profile=fedora28-x86_64  
$ cobbler system list  
test  
$ cobbler system report --name=test  
Name                : test  
TFTP Boot Files      : {}  
Comment             :  
Enable gPXE?        : 0  
Fetchable Files      : {}  
Gateway             :  
Hostname            :  
Image               :  
IPv6 Autoconfiguration : False  
IPv6 Default Device :  
Kernel Options       : {}
```

```
Kernel Options (Post Install) : {}
Kickstart                     : <<inherit>>
Kickstart Metadata           : {}
LDAP Enabled                  : False
LDAP Management Type         : authconfig
Management Classes           : []
Management Parameters        : <<inherit>>
Monit Enabled                 : False
Name Servers                  : []
Name Servers Search Path      : []
Netboot Enabled               : True
Owners                        : ['admin']
Power Management Address      :
Power Management ID           :
Power Management Password     :
Power Management Type         : ipmitool
Power Management Username     :
Profile                       : fedora28-x86_64
Proxy                         : <<inherit>>
Red Hat Management Key        : <<inherit>>
Red Hat Management Server     : <<inherit>>
Repos Enabled                 : False
Server Override               : <<inherit>>
Status                        : production
Template Files                : {}
Virt Auto Boot                : <<inherit>>
Virt CPUs                     : <<inherit>>
Virt Disk Driver Type         : <<inherit>>
Virt File Size(GB)            : <<inherit>>
Virt Path                     : <<inherit>>
Virt RAM (MB)                 : <<inherit>>
Virt Type                     : <<inherit>>
```

La configuration du réseau est la principale raison de la création d'un objet système. Lorsqu'on utilise des profils, on est limité aux interfaces DHCP, mais on peut spécifier beaucoup plus d'options de configuration réseau avec les systèmes.

On va maintenant configurer une seule interface simple dans le réseau 192.168.1 / 24:

```
$ cobbler system edit --name=test --interface=eth0 --mac=00:11:22:AA:BB:CC -
-ip-address=192.168.1.100 --netmask=255.255.255.0 --static=1 --dns-
name=test.mydomain.com
```

La passerelle par défaut n'est pas spécifiée par NIC, donc il faut l'ajouter simplement (avec le nom d'hôte):

```
$ cobbler system edit --name=test --gateway=192.168.1.1 --
hostname=test.mydomain.com
```

--hostname correspond au nom du système local et est renvoyé par la commande "hostname". --dns-name (qui peut être défini par carte réseau) doit correspondre à une image A DNS liée à l'adresse IP de cette adresse. Aucune Interface n'est requise, mais il est recommandé de spécifier les deux. Certaines fonctionnalités avancées (telles que la gestion de la configuration) reposent sur le champ --dns-name pour la recherche des enregistrements système.

Chaque fois qu'un système est édité, cobbler exécute ce que l'on appelle une "synchronisation simplifiée", qui régénère des fichiers critiques tels que le fichier de démarrage PXE dans le répertoire racine TFTP. Une chose qu'il ne fera PAS est d'exécuter des actions de gestion de service, telles que la régénération du dhcpd. Conf et redémarrage du service DHCP Après avoir ajouté un système avec une interface statique, il est judicieux d'exécuter une "synchronisation de cordonnier" complète afin de garantir que le fichier dhcpd.conf est réécrit avec le bail statique correct et que le service est renvoyé.

Utilisation d'un fichier kicksart personnalisé

Création d'un fichier kickstart

/var/lib/cobbler/kickstarts est l'emplacement par défaut des fichiers kickstart.

Créer maintenant un fichier CentOS7.ks en utilisant n'importe quel éditeur de texte.

```
vi /var/lib/cobbler/kickstarts/CentOS7.ks
```

Ajouter le contenu suivant au fichier, puis enregistrer et quitter l'éditeur de texte.

Remplacer l'IP du serveur, le mot de passe root et d'autres détails.

```
# plate-forme = x86, AMD64 ou Intel EM64T
# version = DEVEL
# Configuration du pare-feu
firewall --disabled
# Installer le système d'exploitation au lieu de la mise à niveau
install
# Utilise le support d'installation HTTP
url --url="http://172.168.10.5/cblr/links/CentOS7-x86_64/"
# mot de passe Root
rootpw --iscrypted $1$j9/aR8Et$uovwBsGM.cLGcwR.Nf7Qq0
# Network information
network --bootproto=dhcp --device=eth0 --onboot=on
# Redémarrer après l'installation
reboot
# Information d'autorisation du système
auth useshadow passalgo=sha512
# Utiliser l'installation graphique
graphicalfirstboot disable
```

```
# Clavier système
keyboard us
# Langue du système
lang en_US
# configuration SELinux
selinux disabled
# Niveau de journalisation de l'installation
logging level=info
# Fuseau horaire du système
timezone Europe/Amsterdam
# Configuration du chargeur de démarrage du système
bootloader location=mbrclearpart --all --initlabel
part swap --asprimary --fstype="swap" --size=1024
part /boot --fstype xfs --size=500
part pv.01 --size=1 --grow
volgroup root_vg01 pv.01
logvol / --fstype xfs --name=lv_01 --vgname=root_vg01 --size=1 --
grow%packages
@^minimal@core
%end
%addon com_redhat_kdump --disable --reserve-mb='auto'
%end
```

Utilisation du fichier kickstart

Synchroniser le profil mis à jour sur le serveur Cobbler avec la commande suivante:

```
cobbler profile edit --name=CentOS7-x86_64 --
kickstart=/var/lib/cobbler/kickstarts/CentOS7.ks cobbler sync
```

La commande report doit afficher une nouvelle valeur pour le champ "Kickstart Metadata" :

```
$ cobbler distro report --name=fedora28-x86_64
Name                               : fedora28-x86_64
Architecture                       : x86_64
TFTP Boot Files                    : {}
Breed                              : redhat
Comment                            :
Fetchable Files                    : {}
Initrd                             : /var/www/cobbler/ks_mirror/fedora28-
x86_64/images/pxeboot/initrd.img
Kernel                            : /var/www/cobbler/ks_mirror/fedora28-
x86_64/images/pxeboot/vmlinuz
Kernel Options                     : {}
Kernel Options (Post Install)     : {}
Kickstart Metadata                 : {'tree':
'file:///var/lib/cobbler/kickstarts/'}
```

```
Management Classes      : []
OS Version               : fedora28
Owners                  : ['admin']
Red Hat Management Key   : <<inherit>>
Red Hat Management Server : <<inherit>>
Template Files           : {}
```

Le champ "Kickstart Metadata" (--ksmeta en interne) est utilisé pour les variables diverses, Et contient la variable tree utilisée dans les modèles de kickstart pour spécifier l'URL où se trouvent les fichiers d'installation.

Utilisation d'un miroir yum local

Création d'un miroir Yum

Cobbler permet de mirroring les dépôts afin d'accélérer les installations et les mises à jour des systèmes.

Récupération des données

Pour mirroring le dépôt de base utiliser la commande suivante :

```
# cobbler repo add --
mirror=http://download.fedora.redhat.com/pub/fedora/linux/releases/11/Everything/i386/ --name=f11-i386-everything
```



Il vaut mieux éviter d'utiliser le serveur : `download.fedora.redhat.com`. Préférer un serveur proche, pour connaître la liste des miroirs consulter la liste officielle.

Pour mirroring le dépôt contenant les mises à jour utiliser une commande du type :

```
# cobbler repo add --
mirror=http://download.fedora.redhat.com/pub/fedora/linux/updates/11/ --
name=f11-i386-updates
```

Pour récupérer réellement les données, il faut lancer la commande suivante (attention, c'est assez long !) :

```
# cobbler reposync
```

Utilisation des miroirs

Pour utiliser les miroirs lors de l'installation, il faut l'inclure dans le profile:

```
cobbler profile edit --name=f11-PAE-i386 --repos="f11-i386-updates f11-i386-everything"
```

Pour pouvoir utiliser les miroirs après l'installation, il faut modifier une variable dans le fichier de configuration de cobbler /etc/cobbler/settings :

```
yum_post_install_mirror: 1
```

Il faudra alors mettre à jour régulièrement ces dépôts, pour cela on peut utiliser le service Cron. Voici un exemple de planification de la mise à jour des dépôts à une heure du matin tout les jours :

```
0 1 * * * cobbler reposync --tries=3 --no-fail
```

Afin que les dépôts soient accessibles, il faut démarrer le serveur web apache :

```
/sbin/service httpd start  
/sbin/chkconfig httpd on
```

From:

<http://www.ouarte.garden/> - **dwndoc**

Permanent link:

<http://www.ouarte.garden/doku.php?id=centos:cobbler-overview>

Last update: **2025/02/19 10:59**

